Artykułem o NEPTUNIE 134 rozpoczynamy prezentację mikrokomputerów rodzimej produkcji. Być może planowany cykl publikacji pozwoli potencjalnym nabywcom na świadomy wybór sprzętu, odpowiadającego zarówno ich potrzebom jak i możliwościom finansowym. Prawdopodobnie ubocznym efektem naszych opisów będzie zwiększenie się długości kolejek oczekujących na sprzęt — ale być może otworzy to oczy niektórym decydentom, dla których problem krajowej mikroinformatyki nie istnieje.

NEPTUN zdobył sobie dość szeroką rzeszę entuzjastów wśród naukowców, jako system przydatny w laboratorium. Ewenementem jest przełamanie w tej konstrukcji powszechnego sentymentu do rodziny podzespołów wprowadzonych przez firmę INTEL. Zaowocowało to możliwością wykorzystania oprogramowania, które ze skromnego kontrolera może uczynić NEPTUNA niemal wszechmocnym narzędziem pracy.

NEPTUN 184

Kontroler systemowy NEPTUN 184 zbudowany został na bazie 8-bitowego mikroprocesora NMOS typu 6502. Przeznaczony jest zasadniczo do sterowania aparatury pomiarowej i kontrolnej. System wyposażony jest w interpreter języka BASIC oraz MACROASSEMBLER dla mikroprocesora 6502. Procedury w języku maszynowym mogą być wywoływane z programów napisanych w języku BASIC lub wykonywane jako niezależne programy.

Użytkownik ma do dyspozycji do 32 KB pamięci RAM, możliwość współpracy z dwoma magnetofonami kasetowymi, sprzęg IEC 625 (inaczej nazywany HPIB), sprzęg do drukarki DZM-180 oraz bogaty zestaw dodatkowych modułów funkcjonalnych (m.in. możliwość dołączenia stacji dysków elastycznych, plottera X-Y oraz czytnika i perforatora taśmy papierowej). Wymienione możliwości sprawiają, że obszar zastosowań kontrolera wykracza daleko poza dziedzinę pomiarów i kontroli. Wykorzystanie w systemie mikroprocesora 6502 pozwala zastosować dla NEPTUNA ogromną ilość oprogramowania napisanego dla mikrokomputerów APPLE i COMMODORE (z nieznacznymi jedynie zmianami).

System operacyjny umieszczony został w pamięci typu ROM i zgłasza się bezpośrednio po włączeniu zasilania. System zgłasza się w programie EDYTOR, który umożliwia wprowadzenie, edycję i uruchomienie programu w języku BASIC. Z poziomu EDYTORA możliwe jest przejście do do programu MONITOR, a stąd do programu MACROASSEMBLER. Program EDYTOR jest jednym z głównych atutów, jakimi dysponuje NEPTUN — pozwala na swobodne poruszanie się kursorem po całym obszarze ekranu. Dzięki swobodzie ruchu kursora, a także funkcjom INSERT i DELETE, tekst znajdujący się w polu ekranu może być w dowolny sposób zmieniany. Zmiany dokonywane na ekranie drogą edycji powodują wyłącznie zmiany w pamięci obrazowej. Dopiero naciśnięcie klawisza CR powoduje, że poprawiona linia zostaje wprowadzona do programu. W przypadku, gdy liczba znaków w linii przekroczy 40, następna linia zostaje logicznie dołączona do poprzedniej, umożliwiając korzystanie z logicznej linii osiemdziesięcioznakowej.

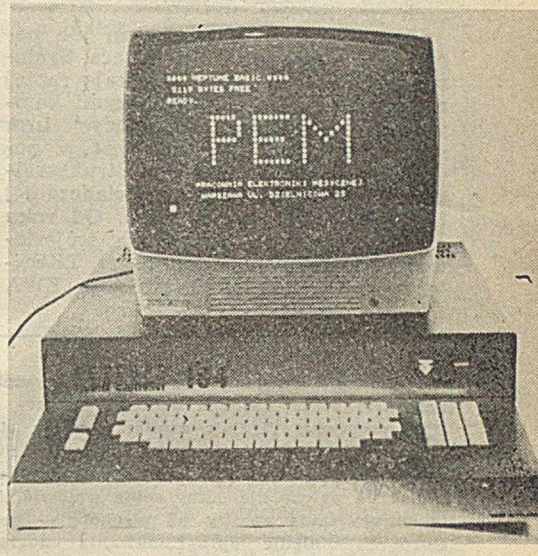
Korzystając z programu plottera ekranowego, można uzyskać na ekranie niemal ciągłe obrazy graficzne (rozdzielczość ekranu wynosi 320×200 punktów). Ekran można więc traktować jako zbiór 64 tys. punktów, z których każdy może być w dwóch stanach — białym lub czarnym. Nie ma możliwości wyświetlania punktów o różnych poziomach szarości, co powoduje, że ewentualne przyłączenie monitora kolorowego nie jest rzeczą prostą.

Mikrokomputer NEPTUN 184

CPU: 6502
RAM: 16 KB
Oprogramowanie: BASIC
Opcje:
dodatkowy RAM 16 KB
dodatkowy EPROM 16 KB lub 32 KB
program ASSEMBLER 6502
sprzęg IEC 625 (HPIB)
sprzęg do drukarki DZM 180
sprzęg V24
sprzęg dysków elastycznych
przetwornik 8×8bit A/C (100 μs)
Cena: 820 tys. zł.
Producent: Pracownia Elektroniki Medycznej
Warszawa-Wawer, ul. Dzielnicowa 25
tel. 12-07-87
Czas realizacji zamówienia: ok. pół roku

Znaki alfanumeryczne definiowane są w matrycy 8×8 punktów. Na ekranie może być wyświetlone 25 wierszy tekstu po 40 znaków w wierszu.

Pamięć obrazu zajmuje 1 KB. Każda 8-bitowa komórka pamięci obrazowej może zawierać kod jednego z 256 znaków, dostępnych w pamięci RAM generatora znaków. Generator znaków z pamięci stałej ROM jest przepisywany do pamięci RAM w czasie inicjacji systemu. Użytkownik ma możliwość wprowadzenia dowolnego własnego generatora znaków.



Klawiatura składa się z 64 klawiszy standardowych oraz 24 klawiszy specjalnych. Każdy ze standardowych klawiszy ma cztery możliwe znaczenia, tak więc bezpośrednio z klawiatury można wygenerować 256 różnych kodów. Podstawowe znaczenie klawiszy oznaczone zostało na klawiaturze. Niestety, specjalne znaczenia nie zostały uwidocznione, co stwarza pewne trudności w użytkowaniu. Wydaje się, że idea, według której zaprojektowano klawiaturę jest bardzo udana, natomiast jej wykonanie pozostawia wiele do życzenia i jest chyba najsłabszą stroną NEPTUNA. Wygląd jej jest mało estetyczny, kształt klawiszy niergonomiczny, a i mechanika nie najlepsza. Naciśnięcie klawisza wymaga pokonania oporu, różnego dla różnych klawiszy i na ogół znacznie większego niż to ma miejsce w nowoczesnych klawiaturach.

Wbudowany zasilacz dostarcza niezbędnych napięć nie tylko dla kontrolera, ale również dla dołączanych na zewnątrz modułów funkcjonalnych. Do odprowadzenia ciepła nie zastosowano wymuszonego obiegu powietrza, co wprawdzie powoduje bezszmerową pracę NEPTUNA, jednakże przy pracy powyżej kilkunastu godzin zdarzają się kłopoty. Wadę tę można wyeliminować w prosty sposób, realizując wymuszony obieg powietrza w okolicy radiatorów zasilacza (wystarczy najprostszy wentylator).

Bardzo dobrze rozwiązana jest współpraca z magnetofonami. Natomiast współpraca NEPTUNA z modułem dysku elastycznego produkcji MERA-KFAP pozostawia jeszcze dużo do życzenia. Producent obiecuje wprowadzenie nowego systemu dyskowego, umożliwiającego łatwą pracę z dyskami elastycznymi z poziomu języka BASIC. Warto tutaj nadmienić, że oprogramowanie i możliwości NEPTUNA

są przez producenta praktycznie ciągle ulepszane, przy czym udało się uniknąć takich przeróbek, które wymagałyby wprowadzania poprawek do starych programów.

Uwagi krytyczne w niczym nie pomniejszają wartości Kontrolera Systemowego Neptun 184 i wynikają raczej z porównania go z najnowszymi komputerami osobistymi przodujących firm światowych. NEPTUN 184 jest w posiadaniu Zakładu Badań Strukturalnych Instytutu Fizyki Doświadczalnej UW od ponad półtora roku. Od roku używany jest do sterowania trójosiowego spektrometru neutronów TKSN-420 przy kanale nr 9 reaktora EWA w Świerku. Taki typ pracy wymaga aparatury o długich okresach międzyawaryjnych, gdyż system musi pracować w nieprzerwanych cotygodniowych cyklach 96-godzinnych w ciągu prawie całego roku. W czasie półrocznego okresu eksploatacji NEPTUNA wystąpiła tylko jedna awaria, usunięta przez producenta w ciągu kilku godzin. W przodujących laboratoriach do sterowania spektrometrów podobnej klasy są używane minikomputery PDP-8, wymieniane ostatnio na komputery PDP-11. Zastosowanie Kontrolera Systemowego NEPTUN 184 pozwoliło na zrealizowanie sterowania eksperymentem „on-line" na poziomie odpowiadającym poziomowi przodujących laboratoriów europejskich.

Wydaje się, że w chwili obecnej NEPTUN jest najatrakcyjniejszym systemem mikroprocesorowym dostępnym na naszym rynku.

WŁADYSŁAW MINOR
Wydział Fizyki
Uniwersytet Warszawski

EO/706/K/84

Artykuł ten został opublikowany w porozumieniu z producentem i przez niego sfinansowany.

W Polsce to legenda, w krajach rozwiniętych zaś — narzędzie pracy każdego dobrego planisty czy zarządcy. Opis programu VisiCalc zamieszczamy z cichą nadzieją, że nie tylko cinkciarze podejmują w Polsce decyzje w oparciu o szczegółową analizę faktów i danych. A już na pewno nie damy sobie wmówić, że ktokolwiek wykonuje takie analizy na piechotę — bez komputera. Rozpowszechnienie oprogramowania mogłoby tu więc uchodzić za papierek lakmusowy pokazujący w jakim stopniu (tak naprawdę, nie w gazetach) realizowana jest u nas racjonalna gospodarka. I tylko bez wykrętów: TABPLAN — czyli polska wersja programu VisiCalc nie podlega reglamentacji i może ją nabyć każda firma za kwotę niewiarygodnie małą w stosunku do korzyści, jakie można uzyskać z racjonalnego gospodarowania.

VisiCalc

Zapewne wielu Czytelników choć raz w życiu mozolnie przygotowywało tabelkę, którą mógłby równie dobrze drukować komputer (rys. 1). To z pozoru łatwe zadanie komplikuje się, jeżeli zażądamy, aby rubryki miały różną szerokość i wysokość lub — co gorzej — by niektóre elementy tabeli były zależne od innych elementów. Problem rozwiązuje natychmiast VisiCalc, program produkowany (ciągle trudno pojąć, że oprogramowanie jest równie dobrą produkcją jak obrabiarki i statki...) przez firmę, która dziś od swego szlagieru nazywa się VisiCorp. A oto podstawy działania i główne możliwości programu, który jest standardowym już wyposażeniem komputerów biurowych.

	Cinkciarz	komputerowy		
Data	27 I 84	3 II 84	10 II 84	17 II 84	Projekcja
Bony (zł)	615.00	640.00	635.00	645.00	655.00
Złoto "2" (tys.zł/g)	5.40	6.00	6.00	6.10	6.20
Złoto PL ($/uncja)	364.10	388.75	391.81	392.17	392.52
Złoto USA ($/uncja)	369.93	387.02	386.25	387.35	388.45

Rys. 1

Spójrzmy na rysunek 2: jest to przedruk z ekranu monitora po uruchomieniu programu VisiCalc. Widzimy przede wszystkim sposób określania wierszy (liczby) oraz kolumn (litery) tabelki. Zwykle za standardową uznaje się matrycę o wymiarach 255×255. Liczba elementów powinna być duża, gdyż niektóre z nich będą pełniły tylko rolę „ozdobników", umożliwiając — na przykład — zatytułowanie tabeli (na rysunku — pierwszy wiersz), oddzielanie rubryk (wiersze 2 oraz 4) lub optyczne „rozstrzelenie" tabeli (wiersze 6 oraz 9). Abyśmy mogli zdefiniować co chcemy umieścić w odpowiednim segmencie macierzy, program został wyposażony w tzw. okienko, które pozwala podglądać wybrany element (w przykładzie jest to element B11). U góry ekranu widać, że element ten został zdefiniowany jako pewna kombinacja innych elementów macierzy. Oczywiście, może tu równie dobrze wystąpić stała liczbowa lub znaki alfanumeryczne tworzące komentarz. Przesunięcie okienka poza widoczny na ekranie obszar powoduje automatyczne przesunięcie całej macierzy — w ten sposób możemy oglądać nie tylko wybrany element, ale i jego otoczenie. Tak stało się przy próbie zdefiniowania elementu F8 (rys. 3). Elementy puste nie są wyświetlane.

```
B11 /F$  (V) (4000/3$B8/B5$31.1)                                    C
Copyright (c) 1980 Software Arts Inc. V1.20Z                        22

     .....A.....B.....C.....D.....
 1.                  Cinkciarz  komputerowy
 2.
 3.Data           27 I 84    3 II 84    10 II 84
 4.
 5.Bony (zł)      615.00     640.00     635.00
 6.
 7.Złoto "2"
 8.(tys.zł/g)       5.40       6.00       6.00
 9.
10.Złoto PL
11.($/uncja)     364.10.    388.75     391.81
12.
```

Rys. 2

Dochodzimy teraz do sedna całego programu. Otóż wyobraźmy sobie, że chcemy zobaczyć co się stanie, jeżeli jeden element macierzy zostanie zmieniony (rys. 4). Wstawiamy wartość 650 zamiast 645. Kolejny rysunek pokazuje zdawałoby się taką samą tabelkę. Ale podkreślone elementy zmieniły się, choć zmiana danych wejściowych dotyczyła tylko jednej pozycji tabeli (por. rys. 5 i 1). W taki to prosty sposób możemy automatycznie uzyskiwać odpowiedź na pytanie: „Co by było, gdyby...?" Pytanie takie stawiają sobie po wielokroć wszyscy dyrektorzy. Jak widać, komputer ułatwia i przyśpiesza znajdowanie odpowiedzi.

By rzeczywiście ułatwiać życie i przyspieszać podejmowanie decyzji VisiCalc posiada wiele opcji. Oto niektóre z nich: możliwość replikacji wzorów określających pewne elementy (w przykładzie — kolumny B, C, D itd. zostały

```
F8  (V) (2$E8-D8)                                                   C
                                                                    22
     .....C.....D.....E.....F.....
 1.Cinkciarz  komputerowy
 2.
 3.    3 II 84    10 II 84   17 II 84   Projekcja
 4.
 5.    640.00     635.00     650.00     665.00
 6.
 7.
 8.      6.00       6.00       6.1.       6.20.
 9.
10.
11.    388.75     391.81     389.15     386.49
12.
```

Rys. 3

E5 (V) 645 C
Value 22

	B	C	D	E
1.		Cinkciarz	komputerowy	
2.				
3.	27 I 84	3 II 84	10 II 84	17 II 84
4.				
5.	615.00	640.00	635.0.	645.00.
6.				
7.				
8.	5.40	6.00	6.00	6.10
9.				
10.				
11.	364.10	388.75	391.81	392.17
12.				

Rys. 4

określone tym samym wzorem, wynikającym z przeliczenia cen). Łatwo możemy wybierać odpowiednie formaty (w przykładzie zastosowano oczywiście format z dwiema cyframi po przecinku, ale gdyby tabela dotyczyła liczby pingwinów na Antarktydzie, to pewnie wolelibyśmy format całkowity). Możemy też zażądać drukowania szerszych lub węższych kolumn. Pozwala to zamieścić mniej lub więcej materiału na danej szerokości papieru. Wreszcie, możemy drukować tylko część tabeli — np. tylko bieżące dane.

Bardziej rozbudowane wersje programu VisiCalc umożliwiają definiowanie całych procedur na zmiennych macierzy, co pozwala np. na dokonywanie regresji liniowej (przewidywanie trendów itp.), a nawet na symulowanie niektórych procesów gospodarczych. Najnowsze wersje programu VisiCalc łączą podstawową ideę tablicy oraz okienka podglądowego z grafiką (wykresy słupkowe danych), połączeniami z innymi komputerami (bazy danych), a nawet z procesorem tekstów. Taki globalny program nazwany LisaCalc (opracowany dla mikrokomputera LISA — patrz INFORMATYKA nr 11, 1983) może współpracować z innymi elementami systemu Lisa (LisaGraph, LisaDraw, LisaList, LisaProject oraz LisaWrite), tworząc najdoskonalsze obecnie biuro w jednej skrzynce!

	Cinkciarz	komputerowy		
Data	27 I 84	3 II 84	10 II 84	17 II 84	Projekcja
Bony (zł)	615.00	640.00	635.00	650.00	665.00
Złoto "Z" (tys.zł/g)	5.40	6.00	6.00	6.10	6.20
Złoto FL ($/uncja)	364.10	388.75	391.81	389.15	386.49
Złoto USA ($/uncja)	369.93	387.02	386.25	387.35	388.45

Rys. 5

W sytuacji, gdy wszystkie stałe są „zmienne", możność przewidywania „co by-gdyby" dzięki jednemu przyciśnięciu klawisza jest pokusą nie do odparcia. Nie będziemy wspominać o liczeniu bilansów, list wypłat i innych typowych we wszystkich biurach zajęciach, które spędzają sen z powiek urzędnikom. A może jednak informatyka jest potrzebna???

JAKUB TATARKIEWICZ

8255 w praktycznych zastosowaniach

Układ 8255, wprowadzony na rynek przez firmę INTEL, jest jednym z najczęściej wykorzystywanych programowanych układów równoległego wejścia-wyjścia. Projektowany był do wykorzystania w systemach sterowanych przez procesor 8080. Można go jednak z powodzeniem zastosować w konstrukcjach zawierających inne procesory firmy INTEL, a nawet procesor firmy ZILOG — Z80. Sformułowanie „programowany" oznacza, że procesor może zdefiniować różne tryby pracy układu, a także wybrać kierunek transmisji niezależnie dla każdego z trzech portów: **A, B** i **C**.

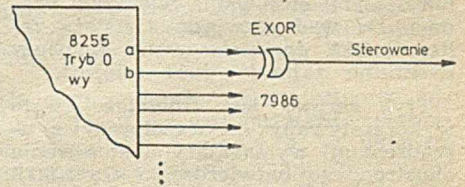
Opis funkcjonalny układu można znaleźć m.in. w [1, 3, 4]. Dlatego też poniżej nie zamieszczamy kolejnego przedruku z katalogu. Skupimy się natomiast na pewnych specyficznych cechach układu, istotnych przy jego stosowaniu. Odpowiednikiem układu 8255 jest produkowany w Polsce MCY 7855.

Wykorzystanie trybu zerowego

W trybie zerowym każdy z portów **A, B** i **C** może być niezależnie zdefiniowany:
— jako wejście służące do badania doprowadzonych sygnałów
— jako wyjście pozwalające wymusić na poszczególnych liniach zadane stany logiczne.
W przypadku portu **C** można również zdefiniować odmienny kierunek przepływu informacji dla dwóch połówek (po 4 bity) portu: „górnej" i „dolnej". Zdarzają się jednak egzemplarze układu, w których odczytanie informacji, z portu **C** powoduje zmianę stanów logicznych w połówce zdefiniowanej jako wyjście. Dlatego też należy unikać takiej konfiguracji portu **C**.

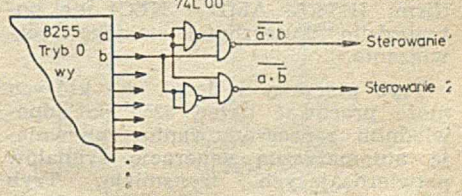
Po wyzerowaniu układu, wszystkie linie portów **A, B, C** znajdują się w stanie dużej impedancji (tzn. układ nie steruje tych linii). Cechą charakterystyczną układów TTL, nie sterowanych na wejściu, jest interpretowanie takiej sytuacji jako stanu wysokiego (ang. High). Tak więc odbiorniki sterowane przez układ 8255, po podaniu w systemie sygnału zerującego RESET, zachowują się tak, jakby były sterowane stanem **H**. Po zaprogramowaniu układu w trybie zerowym, na wszystkich liniach zdefiniowanych jako wyjście, pojawia się stan niski (ang. Low). Dopiero zapisanie danej do portu może zmienić stan linii. Stwarza to problemy przy sterowaniu urządzeń.

Dla zilustrowania problemu rozważmy następujący przykład. Dwie linie wyjściowe wykorzystywane są do sterowania ważnych zaworów, które w pewnych momentach procesu powinny być otwierane i zamykane. Otwarcie zaworu w złym momencie może spowodować eksplozję. Jeżeli stan **H** na wyjściu oznacza zawór zamknięty (a więc wyzerowanie systemu mikroprocesorowego jest niegroźne, bo oba zawory są zamknięte), to zaprogramowanie układu w nieodpowiednim momencie może spowodować, że „wyleciemy w powietrze" zanim system zdąży zapisać bajt danych zamykający zawory. Dla uniknięcia takiej sytuacji stosuje się specjalne rozwiązania sprzętowe.



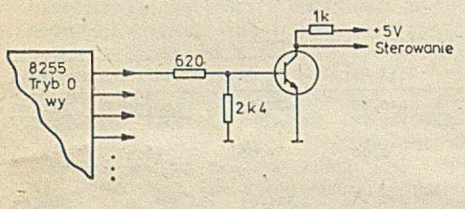
Rys. 1. Na wyjściu sterującym pojawia się stan logiczny 1 tylko gdy $a\bar{b} \vee \bar{a}b$

Najbardziej oczywistą metodą jest zastosowanie do sterowania urządzenia zewnętrznego dwóch linii wyjściowych. Jeżeli na linii podane są różne stany logiczne, to urządzenie jest uruchamiane (np. otwiera się zawór). Na rysunku 1 przedstawiono rozwiązanie wykorzystujące bramkę typu EXOR (ang. EXCLUSIVE OR). Stanem aktywnym wyjściowej linii sterującej jest stan **H**. Rozubudowując nieco układ można za pomocą dwóch linii sterować dwa urządzenia (rysunek 2). Wadą takiego rozwiązania jest brak możliwości załączania obu urzą-



Rys. 2. Stan aktywny na wyjściu 1 pojawia się dla ab = 01, na wyjściu 2 dla ab = = 10

dzeń równocześnie. Ze względu na niewielką obciążalność wyjść układu 8255, zastosowane bramki powinny być typu 74LS00 lub 74L00.



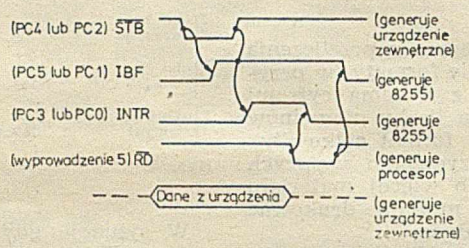
Rys. 3. Stan aktywny na wyjściu pojawia się tylko gdy wyjście portu jest w stanie logicznym 1

Trzecie rozwiązanie (rysunek 3) pozwala wykorzystać każdą linię układu 8255 niezależnie od pozostałych. Jeżeli linia sterująca znajduje się w stanie dużej impedancji, tranzystor jest zatkany i wyjściowa linia sterująca znajduje się w stanie **H**. Odpowiednio dobrana polaryzacja tranzystora zapewnia, że gdy linia wyjściowa układu 8255 znajduje się w stanie **L**, tranzystor nadal pozostaje zatkany. Przejście linii sterującej w stan **H** powoduje odetkanie tranzystora, który wchodzi w obszar pracy w nasyceniu. W związku z tym wyjściowa linia sterująca znajduje się w stanie L. Zastosowanie tranzystora ułatwia ponadto współpracę z odbiornikami wymagającymi sterowania większym prądem. Do wad rozwiązania należy zaliczyć fakt, że zastosowane elementy dyskretne zajmują dużo miejsca.

Tryb zerowy jest również często wykorzystywany do współpracy z urządzeniami wymagającymi stosowania obustronnego potwierdzenia transmisji (ang. hand shake). Obsługa linii kontrolno-sterujących odbywa się wtedy programowo, co upraszcza w znacznym stopniu dopasowanie się do różnych urządzeń.

Dla uniknięcia błędnych stanów należy wykorzystać jedno z przedstawionych wcześniej rozwiązań.
Uwaga: w [1] podano metodę programowego rozwiązania problemu związanego z przejściem wyjść w stan **L** bezpośrednio po zaprogramowaniu układu 8255. Polegać ma ona na zapisie odpowiedniej danej do portu p r z e d zaprogramowaniem układu 8255. Niezbędny jest jednak wybór konfiguracji, w której port **C** zdefiniowany jest jako wejście. Niestety, przetestowanie kilkunastu układów (firm: INTEL, AMD i NEC) nie potwierdziło skuteczności takiego rozwiązania.

Tryb pierwszy i drugi, w których może pracować układ 8255 po odpowiednim zaprogramowaniu, zapewniają automatyczną generację sygnałów potwierdzających transmisję. Tryb pierwszy wykorzystywany jest albo do transmisji wyjściowej, albo do wejściowej, natomiast w trybie drugim port **A** w zależności od sygnałów na liniach kontrolnych służy jako wejście lub wyjście. Ze względu na skomplikowane sterowanie, tryb drugi wykorzystywany jest stosunkowo rzadko. Dlatego też poniżej opiszemy jedynie zagadnienie związane z wykorzystaniem trybu pierwszego. Zasygnalizowane problemy występują również w bardzo zbliżonej formie dla trybu drugiego.

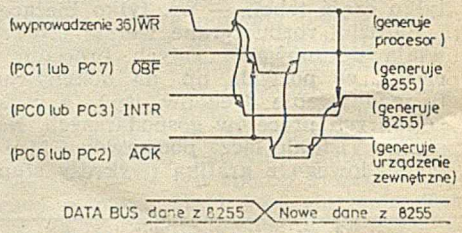


Rys. 4. Wprowadzanie danych z automatycznym generowaniem sygnałów sterujących. Sygnał STB (ang. Strobe) informuje o obecności danych. Sygnał IBF (ang. Input Buffer Full) spełnia rolę potwierdzenia. INTR (ang. Interrupt) — sygnał zgłoszenia przerwania do systemu. RD (ang. Read) — sygnał odczytu danych przez procesor

Na rysunku 4 przedstawiono zależności czasowe przy przekazywaniu informacji przez urządzenie zewnętrzne do systemu. Obecność danych na magistrali komunikacyjnej powinna zostać zasygnalizowana przez stan **L** na wejściu STB. W odpowiedzi układ 8255 automatycznie wymusza stan **H** na wyjściu IBF. Jest to sygnalizacja dla urządzenia zewnętrznego, że może zakończyć cykl, przywracając stan **H** na linii STB. Według wymagań katalogowych, urządzenie zewnętrzne powinno jeszcze przez co najmniej 180 ns sterować linie danych po przywróceniu stanu **H** na linii STB. Jak wskazuje praktyka, czas ten jest podany z dużym zapasem, gdyż poprawne informacje wprowadzane są przez układ, nawet gdy linie danych przestają być sterowane już po ok. 20 ns. Odczyt wprowadzonych danych może nastąpić w dowolnym momencie, gdyż są one pamiętane w wewnętrznym rejestrze układu. Po odczycie, na wyjściu IBF automatycznie wymuszany jest stan **L**, co sygnalizuje, że urządzenie współpracujące może przejść do realizacji następnego cyklu transmisji.

Stan linii IBF może być odczytywany przez procesor w rejestrze wskaźników. Warto jednak ograniczyć korzystanie z tego wskaźnika tylko do specyficznych sytuacji. Nazwa zastosowana przez producenta jest myląca, gdyż wprowadzona do układu informacja nie musi jeszcze odpowiadać rzeczywistej danej. Urządzenie współpracujące spełni wymagania [3], gdy przekaże informację na szynę na moment przed przejściem STB w stan **H**. Poza tym, nawet gdy w chwili przejścia IBF w stan **H** do portu wprowadzane są już prawidłowe dane, zrealizowanie odczytu przez procesor może spowodować przejście linii IBF w stan **L**, zanim urządzenie współpracujące wymusiło ponownie stan **H** na linii STB. Może to spowodować zawieszenie dalszej transmisji.

Wymienione niedogodności wynikające z wykorzystania **IBF** można wyeliminować, odczytując wskaźnik związany z linią **INTR**. Aby było to możliwe, trzeba przed rozpoczęciem transmisji zapisać do rejestru kontrolnego układu 8255 odpowiedni kod odblokowujący wewnętrzny przerzutnik INTE (ang. Interrupt Enable). Należy podkreślić, że „normalny" zapis do portu **C** nie wpływa na stan tego przerzutnika, mimo że teoretycznie zapis do rejestru kontrolnego i do portu **C** oddziaływuje na ten sam bit.



Rys. 5. Wysyłanie danych z automatycznym generowaniem sygnałów sterujących. WR (ang. Write) sygnalizuje zapis danych przez procesor. OBF (ang. Output Buffer Full) sygnalizuje, że dane na szynie mogą być odczytywane przez urządzenie zewnętrzne. ACK (ang. Acknowledge) potwierdza wprowadzanie danych przez urządzenie zewnętrzne. INTR (ang. Interrupt) — sygnał zgłoszenia przerwania do systemu

Na rysunku 5 przedstawiono zależności czasowe przy przekazywaniu informacji z systemu do urządzenia zewnętrznego. Po zapisaniu danych przez procesor przekazywane są one na magistralę komunikacyjną i automatycznie wymuszany jest stan **L** na linii OBF (nieco wcześniej linia zgłoszenia przerwań INTR przechodzi w stan pasywny **L**). W odpowiedzi urządzenie zewnętrzne wymusza stan **L** na linii ACK i rozpoczyna wprowadzanie danej. Układ 8255 automatycznie przywraca stan **H** na linii OBF. Urządzenie zewnętrzne kończy wprowadzanie danej i sygnalizuje to ponownym wymuszeniem stanu **H** na linii ACK. Układ 8255 może teraz przyjąć następną daną co sygnalizowane jest przez stan aktywny **H** na linii INTR.

Stan linii OBF może być odczytywany przez procesor w rejestrze wskaźników. Jednak, podobnie jak przy wprowadzaniu informacji nie wskazane jest wykorzystywanie IBF, tak tutaj wykorzystanie OBF może być powodem błędnego działania. Zapisanie nowej danej przez procesor, gdy OBF przejdzie w stan **H**, może spowodować zmianę informacji podawanej na magistralę komunikacyjną przed zakończeniem wprowadzania przez urządzenie zewnętrzne. Ana-

logicznie jak przy wprowadzaniu danych, lepiej jest wykorzystać wskaźnik odpowiadający linii INTR, po uprzednim odblokowaniu przerzutnika INTE.

Wykorzystanie trybów pracy układu 8255, w których sygnały sterujące generowane są automatycznie, wymaga z reguły zastosowania bardziej lub mniej skomplikowanych układów dopasowujących. Czasami sprowadza się to jedynie do „odwrócenia" sygnałów na którejś z linii, często jednak wymaga zastosowania układów zmieniających relacje czasowe.

Praktycznie, tryb pierwszy i drugi wykorzystuje się w systemach, w których transmisja obsługiwana jest przez przerwania (w trybie zerowym układ 8255 nie generuje sygnałów przerwań). Przy programowym przeglądaniu wskaźników, rozwiązania „automatyczne" nie przyspieszają przebiegu transmisji, a program obsługi jest tylko nieznacznie krótszy.

Tryb drugi stosowany jest w przypadkach gdy magistrala wykorzystywana jest do transmisji „w obie strony", a użycie oddzielnych portów jest niemożliwe. Wykorzystanie trybu zerowego w tym przypadku wymagałoby przeprogramowywania układu przy zmianie kierunku transmisji.

AJP

LITERATURA

[1] An Introduction To Microcomputers. Vol 3, OSBORNE/Mc Graw-Hill

[2] Badźmirowski K, Pieńkos J., Piestrzyński W.: Systemy Mikroprocesorowe. WNT, 1981, str. 230—234

[3] Intel Component Data Catalog 1980. AFN-01300A-1, 3065 Bowers Avenue, Santa Clara, CA 95051

[4] Programowany Równoległy Układ Wejścia/Wyjścia MCY 7855. Instytut Technologii Elektronowej, Al. Lotników 32/46, Warszawa.

- Piotr Lindner jest posiadaczem mikrokomputera SHARP PC 1500. Prowadzi biuro usług matematycznych AKSJOMAT. Niezależnie od tego, zainspirowany przykładem warszawskiego ABAKUSA, próbuje założyć w Łodzi klub mikrokomputerowy. Jeżeli uda się zdobyć odpowiedni lokal (rozmowy w toku), klub powinien rozpocząć działalność od września. Podstawowym sprzętem będzie na razie mikrokomputer ZX81. Poszukiwani są chętni do współpracy — w szczególności osoby dobrze znające ZX81, programowany w języku ASSEMBLER Z80. Kontakt: 91-320 Łódź, ul. Zgierska 142 m. 208; tel. 55-14-45.
- Otrzymujemy wiele telefonów od użytkowników ZX81, którzy poszukują oprogramowania użytkowego (np. ASSEMBLER, FORTH). Prosimy więc osoby dysponujące tego typu oprogramowaniem o udostępnienie go. Oferty chętnie opublikujemy.
- Bieżące informacje prosimy przesyłać — jak najszybciej — pod adresem INFORMATYKI: 00-041 Warszawa, ul. Jasna 14/16 pok. 244; tel. 27-71-40.

W pierwszym wydaniu mikroKLANU zamieściliśmy rozwiązanie umożliwiające wykorzystanie w systemie mikroprocesorowym pamięci dynamicznych. Stosunkowo skomplikowany układ wymagał pewnej wprawy od osób, które chciałyby go zrealizować. W tej sytuacji wielu początkujących konstruktorów decyduje się na stosowanie statycznej pamięci RAM, co niejako automatycznie zmusza do ograniczenia wielkości pamięci o swobodnym dostępie (względy ekonomiczne). Istnieje jednak rozwiązanie pośrednie, czyli tzw. pamięci pseudostatyczne. Eliminują one konieczność multipleksowania adresów, stosowania układów generujących RAS i CAS, a także problem rozstrzygania konfliktu między odświeżaniem a dostępem do pamięci przez procesor.

Opisane poniżej układy trzeba, niestety, sprowadzać za twarde waluty, ale — jeśli wierzyć producentom — są warte żądanej za nie kwoty.

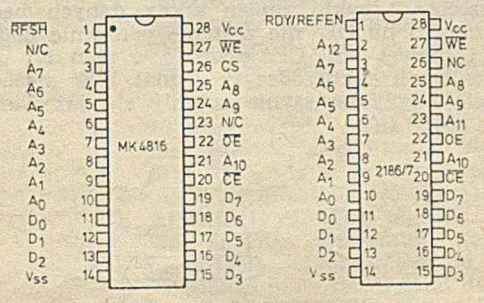
Pamięci pseudostatyczne

Firma MOSTEK, jako jedna z pierwszych, zaproponowała pamięć dynamiczną zintegrowaną z kontrolerem. Układ MK4816 umieszczany jest w 28-wyprowadzeniowych obudowach, co pozwala na wykorzystanie zamiennie z pamięciami ROM i PROM. Jak przystało na nowoczesną konstrukcję, wymaga on pojedynczego napięcia zasilającego (+5 V ± 10%). Układ wyposażony jest w matrycę pamięci o organizacji 2Kx8bitów. Do sterowania pracą układu wykorzystuje się pięć sygnałów: CE/ [1] (ang. Chip Enable), CS (ang. Chip Select), OE/ (ang. Output Enable), WE/ (ang. Write Enable) i RFSH/ (ang. Refresh). Stan jedenastu linii adresowych jest zapamiętywany w układzie z chwilą przejścia w stan 0 sygnału CE/. W tym momencie linia wybierająca układ CS powinna być w stanie 1. Jeśli linia zapisu WE/ pozostanie nieaktywna (stan logiczny 1) zrealizowany zostanie cykl odczytu. Aby dane zostały wpisane na szynę systemową, musi zostać podany sygnał OE/ (stan logiczny 0). Zapis danej do pamięci odbywa się przez wymuszenie stanu logicznego 0 na wejściu WE/.

Odświeżanie zawartości pamięci może być realizowane na kilka sposobów. Każdy zapis lub odczyt powoduje automatycznie odświeżanie jednego z rzędów matrycy pamięci (określonego przez sygnały na liniach adresowych $A_0...A_6$). Sposób ten może być wykorzystany w niektórych rozwiązaniach, gdzie gwarantowane jest odwołanie do każdego z rzędów nie rzadziej niż co 2 ms (np. pamięć obrazowa). W typowych zastosowaniach wygodniej jest wykorzystać metodę automatycznego odświeżania się kolejnych rzędów matrycy, przy każdorazowym podaniu aktywnego impulsu na wejście RFSH/. Jeśli jako procesor wykorzystywany jest układ Z80, to stosowanie pamięci 4816 niczym nie różni się od stosowania pamięci statycznych (oczywiście należy połączyć wyjście RFSH/ procesora z wejściem RFSH/ pamięci).

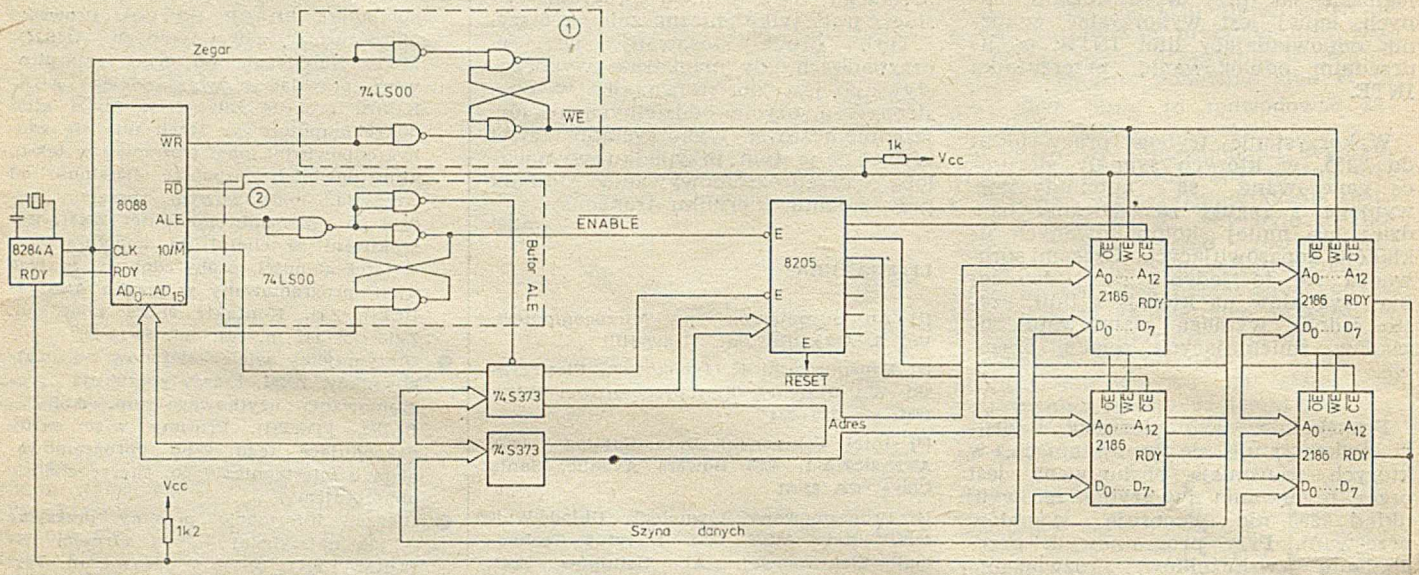
Szczególnie wygodną cechą pamięci jest możliwość samoczynnego odświeżania. Jeżeli na wejściu RFSH/ wymuszony jest przez ok. 20 µs stan logiczny 0, w układzie inicjowane jest samoczynne odświeżanie kolejnych rzędów (mniej więcej co 15 µs kolejny rząd). Cecha ta może być wykorzystana w systemach, gdzie stosowane jest sprzętowe rozwiązanie pracy krokowej, lub też gdy z powodu awarii zasilania niezbędne jest podtrzymanie informacji w pamięci przez zasilanie bateryjne (układ 4816 pobiera tylko 5 mA prądu zasilającego przy pracy w trybie „standby").

Ostatnio firma INTEL wprowadziła na rynek nowy typ pamięci pseudostatycznych. Nazwany on został iRAM (od ang. integrated Random Access Memory) i wyposażony w matrycę pamięci o organizacji 8Kx8 bitów. Podobnie jak omawiany układ firmy MOSTEK, pamięć firmy INTEL produkowana jest w obudowach z 28 wyprowadzeniami. Na rysunku 1 zestawiono rozkład wyprowadzeń dla obydwu układów. Sygnały sterujące pracą układu 2186/7 to: CE/,



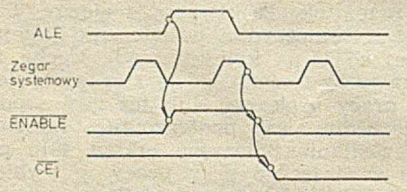
Rys. 1. Topologia wyprowadzeń dla układów MK4816 (MOSTEK) i 2186/7 (INTEL). (Uwaga, powinno być na rysunku RDY/REFEN. Przepraszamy).

[1]) CE/ — zapis ten jest równoważny z zapisem $\overline{CE}$; w całym artykule (poza rysunkami) przyjęto taką notację



Rys. 2. Przykładowe zastosowanie układów 2186 w systemie wykorzystującym mikroprocesor 8088; cztery układy pamięci dają 32 KB RAM

OE/, WE/ i REFEN/ (ang. Refresh Enable). Podobnie jak w przypadku 4816, podanie stanu logicznego 0 na wejście CE inicjuje cykl wykonywania operacji. Bardzo ważne jest, aby sygnał podawany na wejście CE/ był wolny od fałszywych impulsów (ang. glitch) spowodowanych np. przełączaniem dekodera lub buforów, gdyż może to wywołać niekontrolowany cykl operacji. Przykładowa metoda wytworzenia „gładkiego" sygnału CE/ przedstawiona jest na rysunku 2 (blok 2). Sygnał ENABLE/ generowany jest dopiero wówczas, gdy poprawny adres został wpisany w rejestr buforujący. W ten sposób aktywny sygnał na wyjściu dekodera 8205 pojawia się dopiero w chwili, gdy wszystkie linie przyjęły żądany stan. Przebiegi czasowe układu przedstawiono na rysunku 3.

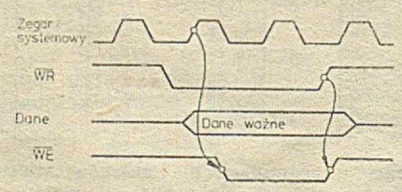


Rys. 3. Przebiegi czasowe dla bloku 2 z rysunku 2

Po rozpoczęciu cyklu przez podanie stanu logicznego 0 na wejście CE/ można zrealizować:
— cykl odczytu (na wejściu OE/ należy wymusić logiczne 0)
— cykl zapisu (na wejściu WE/ należy wymusić logiczne 0)
— cykl „fałszywy" (ang. FMC — False Memory Cycle).

Przy realizacji cyklu zapisu szyna danych będzie sterowana, dopóki na OE/ nie zostanie ponownie podany stan logiczny 1 (nawet jeśli CE/ przeszło w stan 1).

Przy realizacji cyklu odczytu sygnał WE/ może być podany dopiero, gdy na szynie ustaliły się już ważne dane.



Rys. 4. Przebiegi czasowe dla bloku 1 z rysunku 1

W wielu przypadkach konieczne jest więc opóźnienie generowanego przez procesor strobu WR/, zanim zostanie on podany na wejście WE/. Przykładowe rozwiązanie tego problemu pokazano na rysunku 2 (blok 1). Przebiegi czasowe w układzie ilustruje rysunek 4.

Cykl fałszywy jest realizowany wtedy, gdy na żadne z wejść OE/ lub WE/ nie podany zostanie stan logiczny 0. Jeżeli cykl fałszywy jest wykorzystywany dla odświeżenia pamięci, adresy na liniach wejściowych układu powinny być ustalone przed wymuszeniem stanu logicznego 0 na wejściu CE/.

Ponieważ w przypadku pamięci 2186 odświeżanie realizowane jest samoczynnie przez układ, może się zdarzyć, że CE/ zostanie wysterowane w trakcie wewnętrznego cyklu odświeżania. W takiej sytuacji na linii RDY (nóżka 1 układu) pojawia się stan logiczny 0, który może zostać wykorzystany do chwilowego wstrzymania pracy mikroprocesora. W momencie gdy zakończony zostanie wewnętrzny cykl odświeżający, linia RDY ponownie przejdzie w stan 1, co zezwoli mikroprocesorowi na dokończenie operacji. Na rysunku 2 przedstawiono przykładowy sposób realizacji współpracy pamięci 2186 z systemem wykorzystującym mikroprocesor 8088. Wyjścia RDY poszczególnych układów pamięci są sumowane galwanicznie (ang. Wired OR). Układ 8088 pracuje w tzw. minimalnej konfiguracji (jako system jednoprocesorowy). Przy zastosowaniu wersji 5 MHz system może pracować z pełną prędkością, tzn. bez generowania cykli opóźniających przy każdym dostępie do pamięci.

Firma INTEL oferuje też siostrzaną wersję układu pamięci pseudostatycznej o numerze 2187. Różni się ona tym, że na pierwszej nóżce zamiast wyjścia sygnału RDY jest wejście sygnału REFEN/. Przy każdorazowym wymuszeniu stanu logicznego 0, na wejściu REFEN/ inicjowany jest cykl odświeżający, przy czym wewnętrzny licznik generuje adresy kolejnych rzędów. Układ ten jest szczególnie użyteczny przy współpracy z mikroprocesorami generującymi sygnał odświeżania (np. Z80) lub gdy w systemie nie ma możliwości wstrzymywania pracy mikroprocesora.

AJP

LITERATURA

[1] The Designers Guide to iRAMs. Solutions, A Publication of Intel Corporation, March/April 1983
[2] Fallin J. J., Righter W. H.: Designing Memory Systems with the 8Kx8 iRAM. Nota aplikacyjna firmy INTEL AP-132
[3] Memory Data Book and Designers Guide. MOSTEK, 1980.

Informacja o nowym „dziecku" Sinclair'a obiegła warszawskie lobby mikroinformatyczne lotem błyskawicy. Wiadomość o 32 bitach działała niczym Pershing. Ale nie tylko w 32-bitowej arytmetyce rzecz. Wszystko wskazuje na to, że mamy do czynienia z mikrokomputerem o możliwościach porównywalnych z często u nas spotykanym SM-4. Tyle że za cenę bliską połowy miesięcznej pensji Zachodnioeuropejczyka. Niezawodności, a tym bardziej rozmiarów — nie ma nawet co porównywać. Jest jednak coś jeszcze bardziej uderzającego: w roku rozruchu miesięcznika produkcja wyniesie... dwadzieścia tysięcy egzemplarzy. A nie jest to konstrukcja różniąca się od ZX SPECTRUM wyłącznie kształtem obudowy.

Zgoda, trudno porównywać się nam z firmami amerykańskimi — wiadomo, mocarstwo. Lecz tu mamy do czynienia z krajem, o którego trudnościach gospodarczych codziennie donosi prasa PRL. Genialny Mr Sinclair? Być może też.

I jeszcze jedna ciekawostka. Spodziewając się olbrzymiego popytu na całym świecie, Sinclair rozważa możliwości produkowania QL w różnych punktach naszego globu. Mówi się nawet o ... Chinach. Lecz — mimo że składamy ZX81 — nie mówi się o Polsce. Zostawmy to bez komentarza.

Opisawszy poniżej zasadnicze cechy mikrokomputera SINCLAIR QL, nie uznaliśmy sprawy za załatwioną. W miarę napływu bardziej szczegółowych informacji, będziemy się nimi dzielić z Czytelnikami.

Sinclair QL

Londyn, druga połowa stycznia 1984. Sir Clive Sinclair prezentuje swoje nowe dzieło — tym razem nie jest to zabawka dla dzieci czy hobbystów, ale komputer profesjonalny. Swiadczyć ma o tym nazwa QL — Quantum Leap — czyli przeskok o kilka poziomów możliwości użytkowych.

QL skonstruowano z mikroprocesora MOTOROLA 68008, architektura wewnętrzna 32-bitowa, szyna danych 8-bitowa) uzupełnionego układem INTEL 8049 do sterowania klawiaturą. Cztery inne układy scalone (PLESSEY, FERRANTI, NCR i SYNERTEK) realizują funkcje sterowania wyświetlaniem na ekranie, pamięcią i mikrokasetami.

Wymiary: 138 × 46 × 472 mm, waga: 1,38 kg — czyli zachowano formę „klawiatura z rozumem".

Pamięć RAM: 128 K bajtów, rozszerzalna do 640 KB (32 KB zarezerwowane na pamięć ekranu). Pamięć ROM: 32 KB (rozszerzalna do 64 KB) zawiera SUPER-BASIC i system operacyjny Q-DOS (opracowanie własne Sinclaira; system wieloprogramowy, ale nie wielodostępny).

Rozdzielczość ekranu (monitor albo odbiornik telewizyjny): 512 × 256 punktów (przy czterech kolorach) albo 256 × 256 (przy ośmiu). Można wyświetlać po 85 znaków alfanumerycznych w 25 liniach. Klawiatura QWERTY 65-klawiszowa. Dwa „microdrive'y" (jednostki mikrokaset) o pojemności 100 KB każdy.

Cena zestawu podstawowego: 399 funktów szterlingów (1 funt — 1,4 dol. USA).

Oprócz tego dziewięć podłączeń dla dodatkowych układów — np. sześciu mikrokaset, płytki ROM, dwóch sprzęgów RS 232C, 1 M bajta RAM, sieci lokalnej (mogącej połączyć 64 mikrokomputery QL lub SPECTRUM) i jednego lub dwóch drążków sterujących.

Sir Clive nie byłby sobą, gdyby jego konstrukcja nie odróżniała się od innych. Na przykład QL nie ma tak modnej ostatnio myszki ani klawisza kasowania, a klawiatura liczy tylko pięć klawiszy funkcyjnych. W istocie Sinclair starał się zredukować ilość opcji i operacji, by uczynić QL dostępnym dla szerokich rzesz... nieprogramistów.

Podstawowym językiem programowania pozostał BASIC, zwany teraz SUPER-BASICIEM. Poszerzono nieco jego możliwości, m.in. o wykorzystanie okienek, tzn. jednoczesne wyświetlanie informacji z różnych programów (a więc jednak coś z najnowszej mody). Oprócz tego, użytkownik dostaje cztery gotowe programy (wliczone w cenę podstawową!): Q Quill (przetwarzanie tekstów), QL Abacus (modelowanie ekonomiczne), QL Archive (zarządzanie bazą danych) i Q Easel (grafika). Wszystkie mają znormalizowane menu — zwłaszcza Q Quill — do poziomu debiutantów. Natomiast system operacyjny pozwala na jednoczesne wykonywanie dwudziestu programów, z których cztery mogą współdziałać ze sobą.

Jak by tego było za mało, Sinclair przyznał się, że pracuje już nad rozszerzeniem pamięci 0,5 MB, kompilatorem PASCALA, ASSEMBLEREM, PROLOGIEM, emulatorem terminala, sprzęgami: analogowo-cyfrowym, dysku Winchester (!), modemu i drukarki, a także — nad generatorem dźwięków.

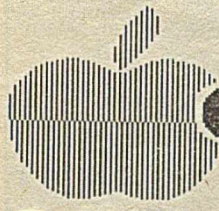
Producent zakłada wytwarzanie tego modelu w liczbie 20 tys. szt. miesięcznie, z perspektywą osiągnięcia w przyszłym roku nawet 100 tys. szt.

Jednocześnie tworzy się klub, a raczej QLUB (QL User Bureau). Składka roczna 35 funtów daje prawo do regularnego biuletynu informacyjnego i otrzymywania aktualnych wersji podstawowych programów.

Oprac.
MAREK SOBCZYK
na podst. 01 Informatique hebdo z dn. 23.01.1984

APPLE II czy nawet III — to już przeszłość; a jednak ciągle sprzedawane są w dużych ilościach. Dlaczego? Przyczyną jest olbrzymia ilość oprogramowania (krążącego również po Polsce), które pozwala nie rozwiązywać każdego problemu od początku. Ten fakt powinni wziąć pod uwagę wszyscy noszący się z zamiarem sprowadzenia sobie mikrokomputera — tym bardziej, że dzięki szerokiej gamie rozszerzeń sprzętowych APPLE jest w stanie zrealizować praktycznie wszystkie zadania, jakie mogą być przydatne dla przeciętnego użytkownika.

Warto też wspomnieć, że mikrokomputer bardzo zbliżony do APPLE II... produkowany jest w Polsce (czegóż nie potrafi prywatna inicjatywa!). Niestety, liczba zamówień przekroczyła możliwości produkcyjne na kilka najbliższych lat. Warsztat rzemieślniczy, nawet najlepiej zorganizowany, nie zastąpi przecież fabryki.



O jabłku opowieść (1)

Kalifornia w 1976 roku była świadkiem narodzin nowego pomysłu: komputera osobistego. Jego konstruktor, Amerykanin polskiego pochodzenia, Stephen Wozniak (wcześniej pracownik firmy elektronicznej ATARI) nadał mu przewrotną nazwę APPLE (jabłko). W ciągu zaledwie kilku lat mikrokomputery te zyskały wielki rozgłos w całych Stanach Zjednoczonych. Sprzedawane są również w Europie Zachodniej, Azji, Australii.

Mimo wciąż rosnącej konkurencji, zainteresowanie JABŁKIEM nie słabnie. Równie dynamicznie rozwija się przy tym wielki rynek oprogramowania oraz dodatkowego sprzętu. Wersja APPLE II posiada obecnie najbogatsze, spośród wszystkich komputerów osobistych, oprogramowanie, a także — wielostronne możliwości dalszej rozbudowy.

Powstają liczne kluby zrzeszające posiadaczy i zwolenników APPLE.

Znane są też liczne związane z firmą periodyki.

Istnieją również ciekawe projekty (niektóre już zrealizowane) wykorzystania mikrokomputera APPLE II w szkolnictwie. Na terenie RFN, w miejscowości Worms, gra on pierwszorzędną rolę w dużym ośrodku szkoleniowym, w skład którego wchodzi technikum, centrum informatyki, gimnazjum i szkoła zawodowa. Pod kierunkiem prof. A. Rissbergera, uczniowie zapoznają się z mikroelektroniką i zasadami informatyki. Podobne szkoły istnieją także w Szwajcarii.

W grudniu ubiegłego roku Musee d'Art Moderne w Paryżu gościł m.in. APPLE II na specjalnej wystawie poświęconej syntezie i twórczości artystycznej obrazów. W połączeniu z plotterem, dwiema kamerami oraz płytą graficzną APPLE przetwarzał i kreował ruchome obrazy.

Centralną częścią APPLE II jest 8-bitowy mikroprocesor 6502 z listą 56 rozkazów, trzynastoma sposobami adresowania, z jednym akumulatorem oraz obszarem adresowania — do 64 KB (16 KB ROM + 48 KB RAM). APPLE wyposażony został w klawiaturę ASCII z 52 klawiszami, w tym klawisze specjalne: ESC, CTRL, RESET, REPT. Do odczytu znaków wykorzystany jest monolityczny dekoder ROM MM5740.

Efekty dźwiękowe można realizować za pomocą zwartego wewnątrz obudowy ośmioomowego głośnika o mocy 0,5 W, sterowanego przez wzmacniacz Darlingtona.

APPLE II zasilany jest przez wysokiej klasy zasilacz wysokoczęstotliwościowy (1,023 MHz) o napięciach wejściowych od 107 do 132 V lub 214 do 264 V oraz napięciach wyjściowych: +5 V, +11,8 V, —5,2 V i —12 V. W zasilaczu zastosowano szereg układów stabilizujących.

Na ekranie monitora standardowo wyświetlane są 24 linie tekstu, po 40 znaków w każdej. Wydruk tekstu możliwy jest w trzech wariantach:
normal — białe litery na czarnym tle
invers — czarne litery na białym tle
flash — wariant migający.

APPLE posiada także możliwości graficzne:
— mała rozdzielczość, czyli 40×40 punktów + 4 wiersze tekstu, 15 kolorów; lub 280×192 punkty, 6 kolorów
— duża rozdzielczość, czyli 280×160 punktów + 4 wiersze tekstu, 6 kolorów; lub 280×192 punktów, 6 kolorów

Pamięć zewnętrzną mogą stanowić:
— zwykła pamięć kasetowa
— dyskietki (ang. floppy disk) — 280 KB
— dyski twarde (ang. harddisk) — 10 MB

Cena: APPLE II (bez dodatków) — ok. 900 dol.; APPLE II + monitor + + stacja napędów do dyskietek (format 5¼") — ok. 1400 dol.

APPLE II dysponuje umiarkowaną dokładnością (9 cyfr). Nie są więc możliwe — często o kluczowym znaczeniu — obliczenia z tzw. podwójną dokładnością. Brak małych liter ogranicza możliwości redagowania tekstów. APPLE II nie przewiduje też możliwości mieszania tekstu z grafiką, a więc np.: opisu wykresów. Żadna wersja APPLE nie pozwala na bezpośrednie podłączenie go do telewizora. Ponadto fakt, iż zasilacz umieszczony został w bardzo bliskim sąsiedztwie głównej płytki, powoduje po dłuższym czasie pracy zbyt duże jej nagrzanie, co w rezultacie może prowadzić do przekłamań w programach.

Większość problemów rozwiązana została przez dodatkowe moduły sprzętowe wraz z towarzyszącym im oprogramowaniem. APPLE zyskał popularność nie tyle za sprawą zalet systemu, lecz również, a może przede wszystkim z powodu swoich nieprzeciętnych możliwości dalszej rozbudowy. Podwójna dokładność możliwa jest po wstawieniu płytki z mikroprocesorem Z80. Może on pracować zupełnie niezależnie lub współpracować z 6502.

Małe litery można wprowadzić w dwojaki sposób. Pierwszy polega na wmontowaniu tzw. **80-Z-CARD.** Karta ta „drukuje" (na monitorze) tekst w 80 kolumnach z dużymi i małymi literami oraz szereg innych znaków, niezbędnych przy redagowaniu tekstu. Drugi sposób to odpowiednie oprogramowanie pozwalające na wprowadzenie małych liter (programy typu: SCREENWRITER).

Program **HRTEXT** lub znacznie droższy **WORDSTAR** (ok. 400 dol.) prócz małych i dużych liter, zestawia tekst z grafiką. Uzyskiwane jest to przez wpisanie punktów tworzących napisy do pamięci obrazowej, tak jakby to była grafika. Stosunkowo łatwo jest więc wprowadzić alfabet grecki czy polski.

Podłączenie do dowolnego czarno-białego telewizora umożliwia dość prosta **UHF-CARD.** Z kolorowymi jest trochę gorzej. Kandydaci do współpracy z APPLE muszą pracować w systemie PAL. Potrzebna jest także specjalna płytka **PAL-CARD.** Podłączenie do kolorowego telewizora uwidocznia niezmiernie bogaty świat grafiki APPLE. Nie najlepiej jest jednak z ostrością tekstu — często występują problemy z przypadkowym jego zabarwieniem.

Konstrukcja APPLE II pozwala na podłączenie od 8 do 16 dodatkowych płytek. I tu pojawia się problem; często bowiem niektóre z nich wykluczają się nawzajem.

Na potrzeby APPLE II opracowano trzy systemy operacyjne: **DOS, UDCSD Pascal, CP/M.** Dwa ostatnie umożliwiają kontakt z innymi mikrokomputerami.

W standardowej wersji oprogramowania występuje interpreter **INTEGERBASIC** lub **APPLESOFT.** Za pomocą tzw. LANGUAGE CARDS można korzystać z następujących języków: **PASCAL, LOGO, APLUS, FORTH, TRANSFORTH, FORTRAN, COBOL, TINY — C, GRAFORTH.**

APPLESOFT — jest pewną modyfikacją języka **BASIC.**
APLUS — zawiera w sobie **APPLESOFT** i stanowi jego znaczne rozszerzenie, jest to „pre-compilator", umożliwiający programowanie strukturalne.
FORTH — język ten został napisany specjalnie na potrzeby mini- i mikrokomputerów. Dla APPLE istnieje w formie interpretera 10—100 razy szybszego niż BASIC.
TRANSFORTH — napisany przez Paula Lutosa jest pewną modyfikacją języka FORTH; uwzględniono w nim specyfikę APPLE.
GRAFORTH — również przystosowany dla potrzeb APPLE II, łączy w sobie wszystkie zalety języka FORTH, jak i wykorzystanie grafiki dwu- i trójwymiarowej wraz z animacją oraz muzyką.
TINY — C — opracowany przez BELL LABS w formie interpretera, umożliwia programowanie strukturalne. (cdn)

PIOTR TYMOCHOWICZ

01-930 Warszawa
ul. Renesansowa 5 m. 10
tel. dom. 35-77-60

W dyspozycji autora znajdują się adresy przedstawicielstw firmy APPLE i klubów użytkowników. Dysponuje on również mikrokomputerem APPLE II i znaczną biblioteką oprogramowania; chętnie więc nawiąże kontakt ze wszystkimi zainteresowanymi.

- **Publikujemy kolejne adresy „agentur" KUM-u. Podane niżej osoby dysponują ankietami klubowymi, zbierają oferty i zapotrzebowania. KOSZALIN:** Jan Marczak — 75-846, ul. Słowiańska 15; tel. d. 333-20, p. 277-91 w. 262. **GLIWICE:** Janusz Najsarek — 44-100, ul. Słowackiego 22 m. 5; tel. p. 31-80-21 w. 1675. **OSTROWIEC ŚWIĘTOKRZYSKI:** Romuald Redlich — 27-400, Osiedle Słoneczne 46/55; tel. p. 236-52 w. 3652. **OPOCZNO:** Krzysztof Surdyk — 26-300, ul. Partyzantów 55 m. 31; tel. d.33-50, p. 34-11 w. 201.
- Informacje o charakterze niekomercjalnym zamieszczamy bezpłatnie. Autorów większych opracowań prosimy o wcześniejsze uzgodnienie z nami tematu i formy prezentacji — coraz częściej bowiem kilka osób opisuje to samo.

— prowadzi
Andrzej J. Piotrowski
(tel. dom. 48-22-85)